



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/761,433	01/16/2001	James Steven Hayko	33262	5201

116 7590 04/15/2005

PEARNE & GORDON LLP  
1801 EAST 9TH STREET  
SUITE 1200  
CLEVELAND, OH 44114-3108

EXAMINER

DELGADO, MICHAEL A

ART UNIT PAPER NUMBER

2144

DATE MAILED: 04/15/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

## Office Action Summary

Application No.

09/761,433

Applicant(s)

HAYKO ET AL.

Examiner

Michael S. A. Delgado

Art Unit

2144

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --  
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

### Status

- 1) ☒ Responsive to communication(s) filed on 25 August 2004.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

### Disposition of Claims

- 4) ☒ Claim(s) 1-31 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-31 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

### Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 16 January 2001 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

### Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
  - ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

### Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)  
Paper No(s)/Mail Date \_\_\_\_\_
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date. \_\_\_\_\_
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: \_\_\_\_\_

## **DETAILED ACTION**

### ***Response to Arguments***

1. Applicant's arguments filed 08/25/2004 have been fully considered but they are not persuasive.

In response to the limitation of a “connection to a network through a communication link”. In Balasurbramaniam’s invention, this limitation is consistent with the communication path (i.e. the internet) that exists between the control server (100) and the user computer (104) (Col 8, lines 50-60).

In response to the argument that “the server resident process through a communication pipe, and interacting directly with an operating system resident on the client” is not taught by the prior art. Balasurbramaniam disclosed a method of using a COM/ActiveX wrapper, which is stored on the client (Col 11, lines 25-30)(Fig 2). The COM provides the platform in which an ActiveX programs can be control via a communication link from a remote server (Col 11, lines 30-55). A COM as defined in Newton Telecom Dictionary, 18<sup>th</sup> Updated and expanded edition by Harry Newton at page 171 is “Component Object Model. COM is Microsoft's cornerstone of its ActiveX platform. (COM is a language independent component architecture (not a programming language). It is meant to be a general purpose, object-oriented means to encapsulate commonly used functions and services. The COM architecture provides a platform independent and distributed platform for multi-threaded applications. COM also encompasses everything previously known as OLE Automation (Object Linking and Embedding). OLE Automation was originally for letting higher level programming languages access COM objects. An object is a set of functions collected into interfaces. Each object has data associated with it.

Art Unit: 2144

The source of the data itself is called the data object. With COM, the transfer of the data itself is separated from the transfer protocol.” The definition of COM as an independent and distributed platform is consistent with operating system that is being claimed by the applicant.

***Claim Rejections - 35 USC § 102***

1. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

2. Claims 1-10 and 12-31 are rejected under 35 U.S.C. 102(e) as being anticipated by US Patent No. 6,701,441 by Balasurbramaniam et al.

In claim 1, Balasurbramaniam teaches about an application provision system for use in a server (Fig 2, 100), the server being connected to a network through a communication link, the network comprises at least one client (Fig 2, 104) and at least one server (Fig 2, 100), the application provision system for providing a requested application to a client, the application provision system comprising (Fig 2) (Col 7, lines 50-55):

a server resident process “COM/ActiveX” for controlling the application provision system and providing a requested application to a client (Col 11, lines 20-30) (Col 9, lines 35-45); and

client-server functionality that is deliverable to the client, the client-server functionality communicating with the server resident process through a communication pipe, and interacting directly with an operating system resident on the client, the communication pipe being a part of the communication link (Col 11, lines 25-30)(Fig 2) (Col 11, lines 30-55) (Col 11, lines 50-60).

In claim 2, Balasurbramaniam teaches about an application provision system according to claim 1 wherein the server resident process includes a common gateway interface (Col 11, lines 5-15).

In claim 3, Balasurbramaniam teaches about an application provision system according to claim 1 wherein the client server functionality is a platform independent executable program attached to a web page (Col 9, lines 35-55).

In claim 4, Balasurbramaniam teaches about an application provision system according to claim 3 wherein the platform independent executable program is a Java applet (Col 9, lines 45-55).

In claim 5, Balasurbramaniam teaches about an application provision system for use in a server (Fig 2, 100), the server being through a communication link, the network comprises at least one client (Fig 2, 104) and at least one server, the client comprising an operating system and a browser, the application provision system comprising (Fig 2) (Col 3, lines 15-25) (Col 8, lines 50-60):

Art Unit: 2144

a server resident process, controlling the application provision system and being able to respond to queries (Col 16, lines 50-67);

a deliverable file that is deliverable to the client, the file including a platform independent executable program “Java” and parameter information (addressing parameter to facilitate search) (Col 9, lines 35-55) (Col 11, lines 5-20); and

a communication pipe within the communications link for providing communications; the platform independent executable program communicating with between the server resident process through the communication pipe and interacting directly with the operating system of the client such that the server resident process is able to perform operations within the operating system of the client (Col 8, lines 50-60) (Col 11, lines 25-30)(Fig 2) (Col 11, lines 30-55) (Col 11, lines 50-60).

In claim 6, Balasurbramaniam teaches about an application provision system according to claim 5 further comprising a database comprising information for the locating of files associated with an application that can be installed on the client (Col 9, lines 55-60).

In claim 7, Balasurbramaniam teaches about an application provision system according to claim 5 wherein the application provision system further comprises;

at least one file associated with at least one application wherein the application is installable on the client (Col 9, lines 55-60); and

at least one web page for downloading to the browser located on the client (Col 9, lines 35-55).

In claim 8, Balasurbramaniam teaches about an application provision system according to claim 5 wherein the file deliverable to the client includes a Java applet (Col 9, lines 50-60).

In claim 9, Balasurbramaniam teaches about an application provision system according to claim 5 wherein the server resident processes includes a common gateway interface (Col 11, lines 5-15).

In claim 10, Balasurbramaniam teaches about an application provision system according to claim 5 wherein the server resident process, client deliverable file and at least one file associated with the at least one application are located on the same server (Fig 2, 100) (Col 11, lines 50-60).

In claim 12, Balasurbramaniam teaches about an web page for delivery from a server to a browser operating on a client (Col 9, lines 35-55), the server (Fig 2, 100) and the client being connected to a network through a communication link, the network comprising at least one client (Fig 2, 104) and at least one server, the server comprising server resident processes "COM/ActiveX" controlling automatic provisioning of an application (Col 7, lines 50-55) (Col 9, lines 35-55) (Col 11, lines 20-30), the web page for use in a system for the automatic provisioning of the application to the client, the web page comprising (Col 9, lines 20-40):

a platform independent executable program communicating with the server resident processes through a communication pipe, and interacting directly with an operating system resident on the client, the communication pipe being a part of the communication link (Col 11, lines 25-30)(Fig 2) (Col 11, lines 30-55) (Col 11, lines 50-60).

In claim 13, Balasurbramaniam teaches about a web page according to claim 12 wherein the platform independent executable program is a Java applet (Col 9, lines 35-55).

In claim 14, Balasurbramaniam teaches about a method of providing a requested application from a server to a client by an application provision system, the method comprising the steps of (Col 4, lines 45-65):

providing a first web page "search program" to a client, the first web page containing at least one link associated with a document (Col 11, lines 5-15);

receiving a request for (link to document) the document by a server resident process "COM/ActiveX", from a client (Col 9, lines 30-40) (Col 11, lines 25-30)(Fig 2) (Col 11, lines 30-55) (Col 11, lines 50-60);

providing a second web page "new executable" to a client, the second web page having client server functionality the client-server communicating with the server resident process through a communication pipe, and interacting directly with an operating system resident on the client and requesting an application to access the requested document (Col 9, lines 30-40) (Col 11, lines 25-30)(Fig 2) (Col 11, lines 30-55) (Col 11, lines 50-60); and

providing the requested application to the client wherein the requested application allows the client to access the requested document (Col 11, lines 20-30).



In claim 15, Balasurbramaniam teaches about a method of providing a requested application according to claim 14 wherein the step of providing the application provides the entire application to the client (Col 11, lines 60-65).

In claim 16, Balasurbramaniam teaches about an method of providing a requested application according to claim 14 wherein the step of providing the application provides selected files associated with the application to the client to update an existing instance of the application on the client (Col 11, lines 60-65).

In claim 17, Balasurbramaniam teaches about a method of receiving a requested application from a server to a client by an application provision system, the method comprising the steps of (Fig 2):

receiving a first web page “search program” from a server, the first web page containing at least one link associated with a document (Col 9, lines 30-40) (Col 11, lines 25-30)(Fig 2) (Col 11, lines 30-55) (Col 11, lines 50-60);

providing a request the document to a server resident process, from a client(Col 11, lines 20-30);

receiving a second web page “new executable software” from a server, the second web page having client server the client-server communicating with the server resident process through a communication pipe, and interacting directly with an operating system resident on the client and requesting an application to access the requested document (Col 9, lines 30-40) (Col 11, lines 25-30)(Fig 2) (Col 11, lines 30-55) (Col 11, lines 50-60); and

receiving the requested application from the server wherein the requested application allows the client to access the requested document (Col 11, lines 20-30).

In claim 18, Balasurbramaniam teaches about a method of receiving a requested application according to claim 17 wherein the step of receiving the application receives the entire application from the client (Col 11, lines 60-65).

In claim 19, Balasurbramaniam teaches about an method of receiving a requested application according to claim 17 wherein the step of receiving the application receives selected files associated with the application from the server to update an existing instance of the application on the client (Col 11, lines 60-65).

In claim 20, Balasurbramaniam teaches about an method for providing a requested application by a server (Fig 2, 100) to a client connected to a network through a communication link, the network comprising at least one client (Fig 2, 104) and at least one server; the server comprising at least one web page for downloading by the at least one client (Col 8, lines 50-60) (Col 9, lines 35-55), the method comprising the steps of:

providing client-server functionality "ActiveX" and a server resident process "COM/ActiveX" on a server (Col 9, lines 25-55);

delivering the client-server functionality to the client during the downloading of a web page to which the client-server functionality is attached the client-server functionality interacting

Art Unit: 2144

directly with an operating system resident on the client (Col 9, lines 35-55) (Col 11, lines 25-30)(Fig 2) (Col 11, lines 30-55) (Col 11, lines 50-60);

receiving a request for an application from the client, through the client-server functionality the client-server functionality communicating with the server resident process through communication pipe, the communication pipe being part of the communication link, the network (Col 8, lines 50-60) (Col 11, lines 50-60); and

providing by way of the server resident process the requested application to the client, through the client-server functionality (Col 11, lines 50-60).

In claim 21, Balasurbramaniam teaches about a method of provisioning of a requested application from a server to a client; the server and the client being connected to a network including at least one client and at least one server, the client operating a browser and an operating system, the method comprising the steps of (Col 11, lines 50-60);

the server and the client being connected to a network, the network including at least one client and at least one server, the client operating a browser and an operating system, the method comprising the steps of (Fig 2):

downloading a first web page containing at least one link, the link, being associated with a document, by the client from the server to which the client is connected and in communications through a communication link (Col 4, lines 45-65) (Col 5, lines 1-15);

selecting the link on the downloaded first web page (Col 4, lines 45-65) (Col 5, lines 1-15);

Art Unit: 2144

selecting the a link on the downloaded web page “navigating” (Col 4, lines 45-65) (Col 5, lines 1-15);

downloading a second web page “sub-window” by the client from the server the second web page comprising wherein-a platform independent executable program; opening a new window of the browser operating on the client (Col 4, lines 45-65) (Col 5, lines 1-15) (Col 9, lines 35-55) (Col 11, lines 20-30);

opening a new window of the browser operating on the client (Col 4, lines 45-65) (Col 5, lines 1-15);

invoking the platform independent executable program the platform independent executable program interacting directly with the operating system of the client (Col 4, lines 45-65) (Col 9, lines 35-55) (Col 11, lines 20-30);

invoking at least one server resident process, the server resident process communicating with the client-server functionality through a communication pipe the communication pipe being part of the communication link (Col 8, lines 50-60) (Col 11, lines 20-65);

scanning the client, by the platform independent executable program to determine if the requested application is installed on the client (Col 11, lines 20-65);

providing the requested application to the client if the requested application is not installed on the client; and invoking the requested application (Col 11, lines 20-65); and

invoking the requested application (Col 11, lines 20-65).

Art Unit: 2144

In claim 22, Balasurbramaniam teaches about a method according to claim 21 further comprising the step of providing code which allows the performance of secure operations within the operating system of the client (Col 10, lines 50-60).

In claim 23, Balasurbramaniam teaches about a method according to claim 21 wherein the step of providing the requested application on the client comprises the steps of:

providing a setup executable associated with the requested application by the server to the client, if the setup executable is absent on the client (Col 11, lines 20-65);

installing the setup executable file on the client, if the setup executable is absent on the client (Col 11, lines 20-65);

invoking the setup executable file (Col 11, lines 20-65);

providing files associated with a requested version of the requested application by the setup executable files from the server to the client (Col 11, lines 20-65); and

installing native components on the client such that the native components do not interfere with the operating system of the client (Col 11, lines 20-65).

In claim 24, Balasurbramaniam teaches about a method according to claim 21 wherein the step of providing the requested application comprises the steps of:

determining if a version of the requested application previously installed on the client is the same as a version of the requested application located on the server (Col 11, lines 50-60); and

installing the requested application on the client if the version of the requested application installed on the client is not the version of the requested application located on the server (Col 11, lines 50-60).

In claim 25, Balasurbramaniam teaches about a method according to claim 24 wherein the step of determining if the version of the requested application installed on the client is the version that is being requested comprises the steps of (Col 11, lines 20-65):

querying an installation footprint of the installed application on the client to determine a location of the files associated with the application installed on the client within a memory of the client (Col 11, lines 20-65);

determining version information for the application installed on the client (Col 11, lines 20-65);

transferring the version information of the application installed on the client to the server (Col 11, lines 20-65); and

comparing the version information of the application installed on the client with version information associated with the version of the application located on the server (Col 11, lines 20-65).

In claim 26, Balasurbramaniam teaches about a method of automatically Provisioning a requested application from a server to a client; the server and client being are elements of a network connected through a communication link the client operating a browser and an operating system, the method comprising the steps of (Fig 2)(Col 11, lines 20-65):

Art Unit: 2144

providing a first web page to the client, the first web page having a link, the link being associated with a document wherein the document is accessible with the requested application (Col 4, lines 45-65) (Col 5, lines 1-15);

receiving a request for a second web page “sub-window” by way of the client selecting the link associated with the document (Col 4, lines 45-65) (Col 5, lines 1-15);

providing the second web page to the client, the second web page comprising a platform independent executable program, the platform independent executable program interacting directly with the operating system of the clients (Col 9, lines 35-55) (Col 4, lines 45-65) (Col 5, lines 1-15) (Col 11, lines 20-30);

receiving communications from the platform independent executable program through a communication pipe, the communication pipe being part of the communication link that invoking a server resident process (Col 11, lines 20-65);

invoking a server resident process (Col 11, lines 20-65);

communicating with the platform independent executable program located on the client to facilitate determination of whether the requested application is installed on the client (Col 11, lines 20-65);

providing files associated with the requested application to the client for installation of the requested application on the client if the requested application is not installed on the client (Col 11, lines 20-65);

transferring files associated with the requested application to the client for installation of the application on the client (Col 11, lines 20-65); and

invoking the requested application (Col 11, lines 20-65).

In claim 27, Balasurbramaniam teaches about a method according to claim 26 wherein the step of providing files comprises the steps of:

providing the setup executable program associated with the requested application by the server to the client, if the setup executable does not exist on the client (Col 11, lines 20-65);

installing the setup executable program on the client, if the setup executable program does not exist on the client (Col 11, lines 20-65);

invoking the setup executable program (Col 11, lines 50-65);

providing files associated with a requested version of the requested application by the setup executable program from the server to the client (Col 11, lines 50-65); and

installing native components on the client such that the native components do not interfere with the operating system of the client (Col 11, lines 50-65).

In claim 28, Balasurbramaniam teaches about a method according to claim 26 wherein the step of providing files comprises the steps of:

comparing version information of the requested application installed on the client with version information of the requested application located on the server if it was determined that the requested application was installed on the client (Col 11, lines 20-65); and

transferring files associated with the requested application to the client for installation of the requested application on the client if the version of the requested application on the client is not the same as the version of the application located on the server (Col 11, lines 20-65).



In claim 29, Balasurbramaniam teaches about a method according to claim 26 wherein the step of providing files comprises the steps of:

querying an installation footprint of the installed application installed on the client to determine a location of files associated with the application installed on the client within a memory of the client (Col 11, lines 20-65);

determining version information for the application installed on the client (Col 11, lines 50-65);

transferring the version information of the application installed on the client to the server (Col 11, lines 50-65); and

comparing the version information of the application installed on the client with version information associated with the version of the application located on the server (Col 11, lines 20-65).

In claim 30, Balasurbramaniam teaches about a computer readable memory element storing the instructions or statements for use in the execution in a computer of a method of automatic provisioning of a requested application from a server to a client; the server and client being elements of a network connected through a communication link the client operating a browser and an operating system, the method comprising the steps of (Col 8, lines 50-60) (Fig 2):

providing a first web page to the client, the first web page having a link, the link being associated with a document wherein the document is accessible with the requested application (Col 4, lines 45-65) (Col 5, lines 1-15);

Art Unit: 2144

receiving a request for a second web page “sub-window” by way of the client selecting the link associated with the document (Col 4, lines 45-65) (Col 5, lines 1-15);

providing the second web page to the client, the second web page comprising a platform independent executable program, the Platform independent executable program interacting directly with the operating system of the client (Col 4, lines 45-65) (Col 5, lines 1-15) (Col 9, lines 35-55) (Col 11, lines 20-30);

receiving communications from the platform independent executable program through a communication pipe, the communication pipe being part of the communication link (Col 11, lines 20-65);

invoking a server resident process (Col 11, lines 50-65);

communicating with the platform independent executable program located on the client to facilitate determination of whether the requested application is installed on the client (Col 11, lines 20-65);

providing files associated with the requested application to the client for installation of the requested application on the client, if the requested application is not installed on the client (Col 11, lines 50-65);

transferring files associated with the requested application to the client for installation of the application on the client (Col 11, lines 50-65); and

invoking the requested application (Col 11, lines 50-65).

In claim 31, Balasurbramaniam teaches about an electronic for use in the execution in a computer of a method for automatic provisioning of a requested application from a server to a

Art Unit: 2144

client; the server (Fig 2, 100) and client (Fig 2, 104) being elements of a network connected through a communication link, the client operating a browser and an operating system "COM", the method comprising the steps of (Fig 2) (Col 8, lines 50-60) (Col 11, lines 20-30):

providing a first web page "main window" to the client, the first web page having a link, the link being associated with a document wherein the document is accessible with the requested application (Col 4, lines 45-65) (Col 5, lines 1-15);

receiving a request for a second web page "sub-window" by way of the client selecting the link associated with the document (Col 4, lines 45-65) (Col 5, lines 1-15);

providing the second web page to the client, the second web page comprising a platform independent executable program, the platform independent executable program interacting directly with an operating system of the client (Col 4, lines 45-65) (Col 5, lines 1-15) (Col 11, lines 20-30);

receiving communications from the platform independent executable program through a communication pipe, the communication pipe being part of the communication link (Col 8, lines 50-60) (Col 11, lines 50-65);

invoking a server resident process (Col 11, lines 50-65);

communicating with the platform independent executable program located on the client to facilitate determination of whether the requested application is installed on the client (Col 11, lines 20-30);

providing files associated with the requested application to the client for installation of the requested application on the client if the requested application is not installed on the client (Col 11, lines 20-65);

Art Unit: 2144

transferring files associated with the requested application requested by the client for installation of the application on the client; and invoking the requested application (Col 11, lines 20-65); and

invoking the requested application (Col 11, lines 20-65).

### ***Claim Rejections - 35 USC § 103***

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

4. Claim 11 is rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent No. 6,701,441 as applied to claim 5 above, and further in view of US Patent 6,606,744 by Mikurak.

5. The factual inquiries set forth in *Graham v. John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

In claim 11, Balasurbramaniam teaches all the limitation but does not explicitly teach about implementing the functionality of software update across different servers.

The concept of using more than one server to implement a service is well known in the art as disclosed by Mikurak (Col 172, lines 20-35). It would have been obvious at the time of the invention for some of ordinary skill to use more than one server to implement a service in order to prevent network congestion.

In an application that involves a large number of remote clients, there will always be network bottleneck. Bottleneck causes the network to be congested and the server will be overburden with the many requests from the clients. This will lead to down time, which is undesirable in business operations. By using more than one server, the burden of servicing many clients is shared across the different servers. By load balancing, the network is more robust and down time is reduced to a minimum.

### ***Conclusion***

6. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Art Unit: 2144

7. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

US Patent No. 6,074,434 by Cole et al., teaches about a selection of code updates, data updates or new data for client.


US Patent No. 6,314,565 by Kenner et al., teaches about a system and method for automated identification, retrieval, and installation of multimedia software components.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Michael S. A. Delgado whose telephone number is 703-305-8057. The examiner can normally be reached on 7.30 AM - 5.30PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, WILLIAM A CUCHLINSKI JR can be reached on (703)308-3873. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

  
MD

  
WILLIAM A. CUCHLINSKI, JR.  
SUPERVISORY PATENT EXAMINER  
TECHNOLOGY CENTER 2000